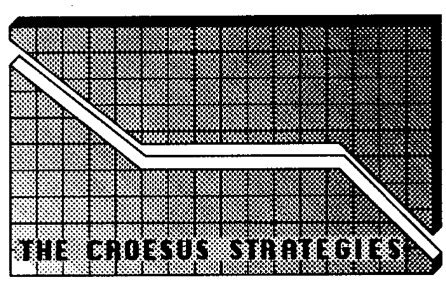
DTIC QUALITY INSPECTED 4

PLEASE RETURN TO:

BMD TECHNICAL INFORMATION CENTER
BALLISTIC MISSILE DEFENSE ORGANIZATION
7100 DEFENSE PENTAGON
WASHINGTON D.C. 20301-7100



19980309 268

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

U3615
cy 1

Accession Number: 3615

Publication Date: Jun 01, 1992

Title: Croesus Strategies, New Approaches to Fielding Information Systems Technologies

Personal Author: Loescher, M.S.

Corporate Author Or Publisher: Director, Space & Electronic Warfare (OP-094) OCNA, The Pentagon, Wash

Comments on Document: Third paper in a Trilogy

Descriptors, Keywords: Navy C4I Program Strategy Acquisition System SEW Architecture Croesus Sonata Copernicus Technology Operation Information Pyramid Warfare Combat

Pages: 00031

Cataloged Date: Jul 09, 1992

Document Type: HC

Number of Copies In Library: 000002

Record ID: 24275

THE CROESUS STRATEGIES

New Approaches to Fielding Information Systems Technologies

Final Draft

Lieutenant Commander M.S. Loescher

June 1992

Director, Space and Electronic Warfare (OP-094)
Office of the Chief of Naval Operations
Room 5B740, The Pentagon
Washington, D.C. 20301

PREFACE

This paper concludes a trilogy that began with *The Copernicus Architecture* in August 1991. Copernicus addressed the need and provided an approach for a new C^4I strategy in the post-Cold War. But C^4I is only a supporting part of a new kind of warfare, the doctrine and technology of which were addressed in the second paper, *Space and Electronic Warfare*, in April 1992. *The Croesus Strategies* proposes to solve three seemingly difficult problems that clear the way to build Copernicus and the other subsystems needed to conduct SEW. In that sense, all three papers are related. Metaphorically, they are different movements of a sonata. In logical sequence, the first movement, Space and Electronic Warfare, addresses SEW definition, doctrine, technology, and techniques. Copernicus, the second movement, describes C^4I as a means of conveying information so critical to the conduct of SEW. This paper outlines the three programmatic strategies by which we will proceed.

The strategies of Croesus, adopted within OP-094, should be viewed as evolving policy. They are three. Croesus first describes a methodology to realign Navy's existing SEW programs into a new structure suitable for the post-Cold War. A common model is developed that allows operators, resource managers, and engineers to achieve balance and a common solution in their efforts. Second, we propose a new means for Navy (and for others who wish to use it) to acquire better and lower cost electronics and computer technology faster, while remaining consistent with DoD acquisition policy. And third, we bring industrial assembly line techniques into Navy in order to converge divergent processes and recoup money for the taxpayer at each step.

These are bold statements. Conventional wisdom has it that because of budget cuts, DoD will be able to buy less. That seems true on the surface. But all conventional wisdom begins first with an unconventional view. We are reminded of the story of financier Bernard Baruch, who was approached by a distraught colleague on the day of the 1929 stock market crash. The colleague wrung his hands and said it was the worst day in economic history. Baruch smiled and replied, "Not for buyers."

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE | | PAGE

PROLOGUE

Before the fifth century BC and prior to the dramatic rise of Athenian civilization, the classical Greeks began to settle the coastal areas of the Mediterranean, Aegean and Black Seas. Seeking to escape the political upheavals and repeated invasions following the decline of Mycenae, these settlers inaugurated the Age of Colonization (750 to 550 BC.) This age was one of expanding trade, growing cities, and increasing literacy and artistic output. Each of these trends marked the end of the Dark Age and foreshadowed the brighter times ahead.

Once established, the independent new city-states (for so the colonies became) developed into markets for products from home and outlets for the exchange of goods with the surrounding areas. Cities specializing in textiles, armaments, pottery, or shipbuilding had a promising future and the grain, fish and slaves sent back from the colonies helped to sustain growing urban populations. By this process, local self-sufficiency gave way to overseas involvement.

Croesus, last of the Lydian Kings, ascended the throne at the end of the Colonial Age, about 560 BC. His reign constituted the most prosperous period of Lydian history. Under him Lydia reached the height of her power, acquiring empire status. The key was money.

INTRODUCTION

Croesus manufactured money; it was his stock in trade. In doing so, he provided the impetus to create a new economic system that revolutionized trade in the Greek world. Croesus made his money in the form of exceptionally pure, and therefore exceptionally valuable, gold and silver coins, which became the basis for trade across the then known world.

The result was a new prosperity. Where previously there had been no economic system, all trade in the eastern Mediterranean became linked. The small country of Lydia, itself inconsequential in military power, agricultural production, or population, rose to prominence on the strength of a new idea. From this idea, all countries great and small were brought into a new world.

We in Navy are poised similarly to capture economic opportunities and build a new programmatic world for Government. As in Croesus' world, however, a new construct, one perhaps not obvious at first glance, will be required.

We are in the midst of the Third Industrial Revolution; our children's grandchildren will learn in school that computers were the harbingers. Information will become a commodity that permeates and strengthens the foundation of everyday activities—from music to commerce to science—for the true value of computers lies in their capacity as the universal machine.

Computers can be made to appear however the user wishes through the simultaneous stacking of the machine's processes. A humanities student working with computer-drawn art on a screen is assisted by the syntax of the operating system and other languages, which make use of binary mathematics that in turn reflect the physics of electrons. Like life itself, computers allow us to experience reality on several planes simultaneously. Artist, engineer, mathematician, and physicist come to a common machine and take away different views.

The impact of the information age is now everywhere apparent. The pace of progress in science, medicine, economics, and a hundred other disciplines will be accelerated beyond our current comprehension. And, like the impact of the printing press 500 years ago, this acceleration will be experienced worldwide, having a profound effect—perhaps eventually creating a new global culture, with all the gains and losses such change connotes.

War and the conduct of it will be affected. At its simplest, the advent of SEW is the reflection and recognition of that change, marking the achievement of the means of conducting warfare in the information age. As in all ages, such developments pose risks and opportunities. We have addressed risks and opportunities of warfare in this new world in a previous paper entitled *Space and Electronic Warfare.*[1] Here we address the implications of these developments in the fielding of information systems.

In electronics production, design, sales, and distribution, times also have changed. Industry is seeking markets outside the military. DoD no longer is driving research and development. In 1962, the U.S. Navy was responsible for over 50 percent of the nation's research and development expenditures on electronics. Thirty years later, Navy is less than five percent.

At the same time, computing and electronics technology hardware, as we have

[1] *Space and Electronic Warfare, A Navy Policy Paper on a New Warfare Area,* (draft), OP-094, April 1992.

seen, is proceeding at a phenomenal rate of change. Computers use five percent of the electrical power in the U.S. today. Witness also Figure 1, from a Mitre study of electronics in DoD systems, which shows the electronic part discontinuation notices from industry to DoD between 1986 and 3rd Quarter 1990.[2] We may only be beginning to see a similar but potentially much more rapid acceleration in new software. Figure 2, from a companion Mitre study of software,[3] shows the rise in expenditures within DoD for software—software that within Navy, we now believe averages eight to 10 years old. Much of our software, therefore, predates the proliferation of personal computers.

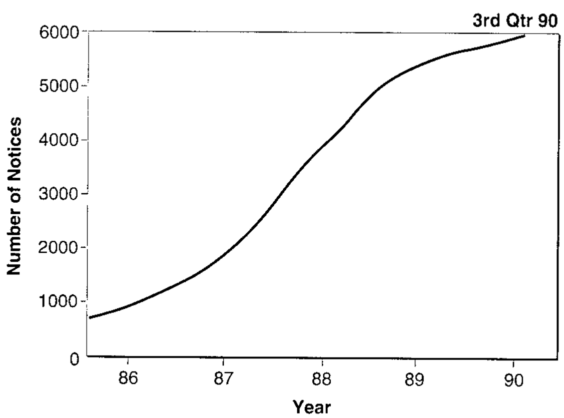


Figure 1. Electronic Discontinuation Notices From Industry

When we first consider such change, we can only worry. In an era of declining budgets, when technological generations are less than a sailor's tour length, how can Navy stay on the cutting edge of technology? How can we operate against an enemy who can quickly buy such technology off-the-shelf at greatly reduced cost? It is perhaps not unreasonable to conclude that in the arms race, the U.S. and its allies stayed ahead of the Warsaw Pact not

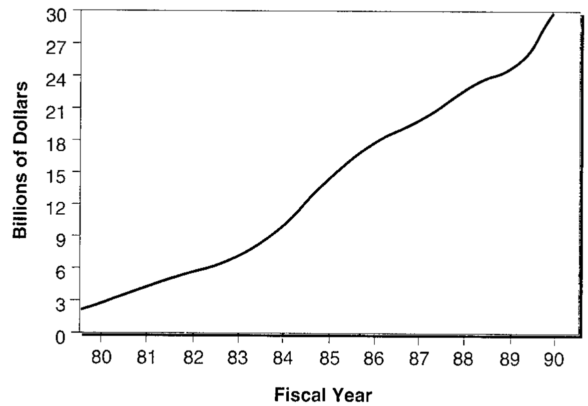


Figure 2. DoD Software Expenditures

only because of the technology itself, but also because acquisition of technology in the Pact took longer than in NATO.

That is no longer true when state-of-the-art is off-the-shelf. While the news media characterized Desert Storm as a high-technology war, professionals knew that military technology—especially computing—was actually low technology when compared with that available today in industry.

Yet, to balance our worries—which certainly are legitimate—there are at least three important opportunities:

- First, in the 1990s, the taxpayer should no longer have to pay for the high cost of technology. Today, much, if not most, of the research and development in computing and electronics needed for the military is being accomplished in industry—by industry;

- Second, we can now solve many problems that have been extremely difficult in the past. Examples abound: in communications, data processing,

[2] From *Modernizing Electronics in DoD Systems,* Dr. Barry M. Horowitz, MITRE Corp., August 1990.
[3] From *The Importance of Architecture in DoD Software,* Dr. Barry M. Horowitz, MITRE Corp., July 1991.

data display, sensor fusion, electronic combat devices, guidance systems—we are reaching levels of such technological sophistication that operational problems that seemed so difficult 10 years ago are now much easier. So much easier in so short a time that, in some cases we sometimes refuse to recognize the opportunities; and

- Third, it follows from the first two, that if we can devise a means to capitalize on these developments, Navy can have higher technology both faster and at lower cost.

This last conclusion sounds wistful in Government circles. Yet in our homes, we see it daily. From televisions to computers, from CD players to watches, technology is better, more available, and more economical each year. And as it was for Croesus, whose idea fostered a new economic system—the time is clearly right: budgets are declining; technology is soaring; the threat is changing.

It is a matter of devising a new process to capture these opportunities.

We call this paper *The Croesus Strategies* because it proposes a new process molded from three steps, intended to be implemented sequentially. The first is the notion of "Pyramidal Programming." The second is the idea of "Cyclical Acquisition." The third, called the "Fleet Assembly Line," brings industrial techniques to bear for Government use.

THE PURPOSE OF ACQUISITION SYSTEMS

The purpose of an acquisition system is to relate technological means to operational ends. Even more so than military warfare, naval doctrine is inextricably tied to technology. Technology and doctrine are components of the same cycle: one fuels the other. Brilliance in naval command invariably is rooted in masterful understanding of naval technology. It has always been so.

The idea that Operational Requirements (ORs) can be divorced from the detailed understanding of their implementing technology is surely an engineer's, not an operator's. "Tell us what you want, and we'll build it" has a tinny ring. Moreover, the assumption on which it is based is neither born out in practice nor in history. The Japanese Long Lance torpedo of World War II made night-fighting destroyer tactics possible; technical understanding and proliferation of the American surface radar six months later erased that advantage. Submarines bred attack submarines. Carrier battle forces bred Soviet Naval Aviation massed-attack tactics. In practice, the best ORs come from operators who understand technology in detail and who can, in their mind's eye, envision the new tactics it makes possible.

In an acquisition system, linkage between technological means and operational ends is achieved through a programmatic structure. At its simplest, the ideal acquisition system must have several recognizable attributes. Foremost, it must preserve relationships between technology and operations throughout the programmatic structure so that the consequence of change may be seen and quantified. Second, money *is* an object; the system must be efficient: the inputs and outputs of the system must be both definable and measurable. Finally, the system must be capable of smooth acceleration and deceleration as demand for output or change in input occurs.

Our current system is like a cube. At the bottom of the cube are enabling technologies—our means. See Figure 3. The number of enabling technologies (significantly, we may not call them "building blocks") remain a vestige of the last two decades, when system engineers built from a menu of hardware and software the size of automobile parts catalogs. We should not berate ourselves too much over this: it was only 1984 that the personal computer came into widespread use. And, today's trend is encouraging if we can capitalize on it. Open-systems architectures, standard protocols, and hardware and software standards are reducing the catalog size significantly.

There is another trend, which we have already seen—the migration of electronics research and development from the military to industry. The impact of these two trends is to make the bottom of the cube smaller. There are fewer and more universal building blocks, making the construction of computer and electronics systems easier, faster, cheaper, and more uniform.

The top of the cube similarly has changed. SEW is the consequential doctrine—a direct military by-product—of the information age. Like the information age, it reflects a cultural change rather than simply the sum of its components—electronic warfare, command and control, communications, surveillance—which have been ends unto themselves since World War II.

This thought is worth pursuing for a moment; there are both military and civilian precedents for cultural change heralded by technology. In the civilian world, they are manifest. Henry Ford's achievement was less the Model T than the automobile culture with its roads, refineries, showrooms, and automobile financing. Thomas Edison's light bulb led to the electric dynamos and power distribution systems of Samuel Insull. There is an identifiable pattern: inventions lead to system building.

In the military, where inventions also lead to systems, the most pertinent example is what is now called C^4I. Figure 4 shows its development from the 1942 Solomons campaigns to the battle of the Philippine Sea in June 1944. The contributing ingredients were three. In surveillance, individual "black

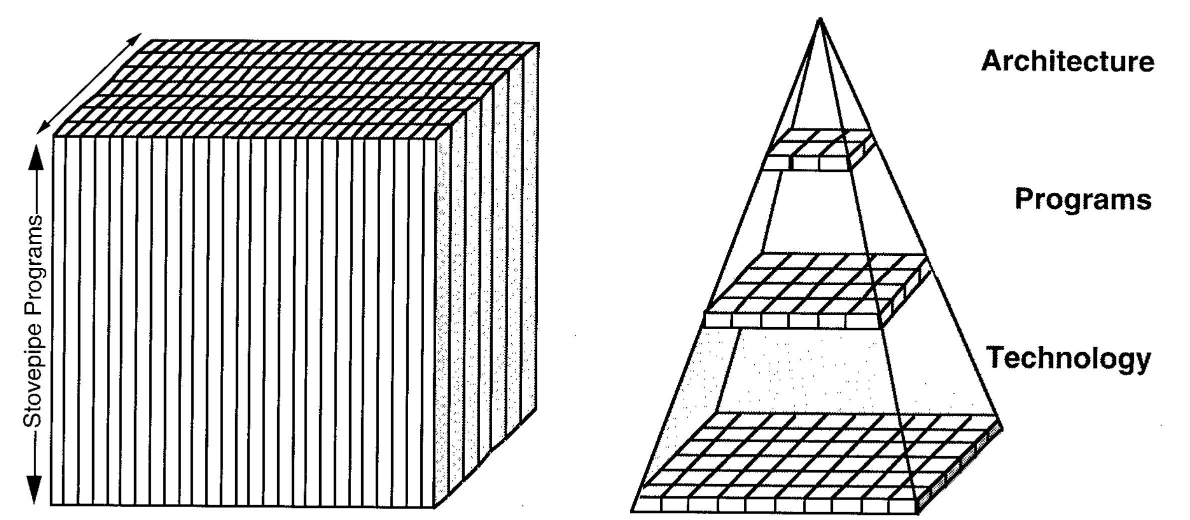


Figure 3. Stovepipe POM Versus Building Block POM

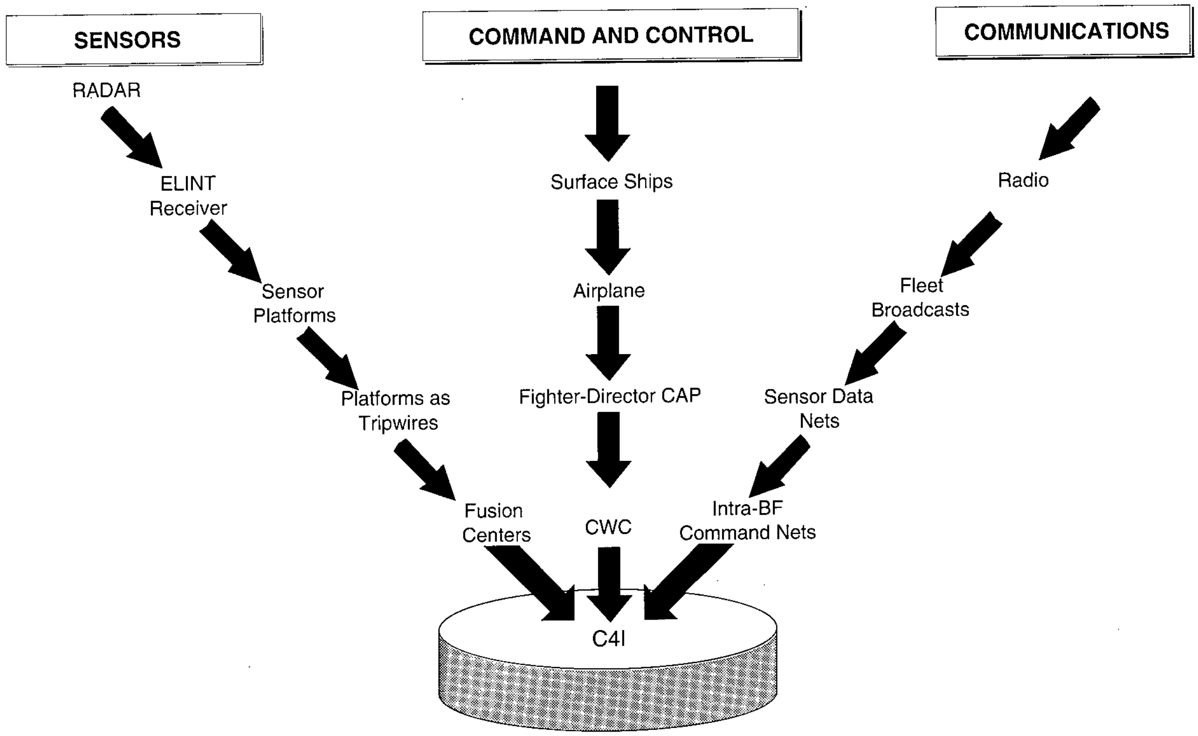


Figure 4. Origins of Modern Naval C^{4}I in World War II.

boxes"—radars, ELINT receivers, COMINT receivers, direction-finders, sophisicated military cameras—brought into theater from the laboratories and universities were stepwise first applied to platforms, platforms then were deployed as tripwires, and finally warning from tripwires consolidated in Pearl Harbor fusion centers. Communications began as voice, evolved to messages, then message broadcasts, then specialized networks. In the middle, a command and control doctrine grew out of lessons-learned in three years of war. By the Battle of the Philippine Sea in June 1944, C^{4}I—simultaneous offense and defense of battle forces, aided by organic and non-organic sensors, supported by structured communications networks—was invented.

So it is with the advent of SEW; it is a new way of conducting warfare. SEW, therefore, sharpens our programmatic cube to a point at top, bringing the previous ends into a new unified, doctrinal construct. This development, coupled with the reduction in the number of technological building blocks, forces us to build a new acquisition model as well, for both ends and means have changed. Necessarily, then, must the middle.

PYRAMIDAL PROGRAMMING

Consider a pyramid, in which the strategic objective of SEW at the top is achieved by the production and placement of individual building blocks that make up its base. In a Government model, once again, the middle tiers represent the programmatic structure that relates means to ends. See Figure 5.

In such a model, which we shall call the Croesus Pyramid, there are three sets of tiers.

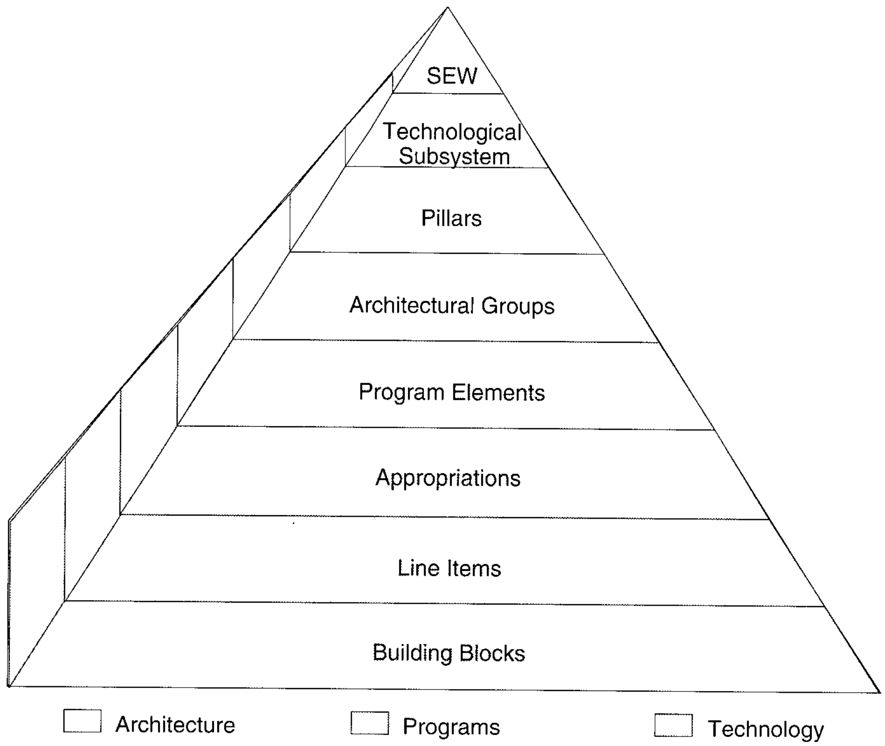


Figure 5. The Croesus Pyramid

Architectural tiers describe ends in layers of increasing detail as we traverse down the pyramid toward the bottom. *Programmatic tiers* below identify the taxonomy of funding: program elements, appropriations, and lines. *Technological tiers* at the bottom represent building blocks and technological systems of related building blocks.

As in the ideal system model, the pyramid must have three attributes:

- Stability (The preservation of relationships among architecture, programs, and technology so that change in any one may be quantified in the others);
- Efficiency (Inputs and outputs defined and measurable); and
- Flexibility (The capability to be smoothly accelerated and decelerated).

In the Croesus Pyramid, *stability* results from consistent vertical and horizontal relationships. The ideal is articulated in the doctrine of SEW—the strategic objective at the top, below which are programs and eventually a finite set of building blocks that share consistent, clearly articulated, relationships. Like the universal machine, operators, resource managers, and engineers can come to a common model and take away different, but related and consistent views. At the foot of the Pyramid are building block tiers that define, in manufacturing terms, the product line of the system. This is the output of the model from an engineer's point of view. From the operator's perspective, the output of the model is a unified SEW technological system that can be used to conduct SEW. The resource manager sees yet another—a coherent and related set of programs that lead the SEW system above through the development of the engineer's building blocks below.

Horizontal Relationships

In a consistent and quantifiable relationship to the SEW objective at the top, the elements on any given *horizontal tier* will be linked *operationally* with other elements on the same tier. By looking left and right across a tier, we can see how elements fit with one another because they share a common denominator on the Pyramid—architecture. Subsystems are related to subsystems in the same way that pillars are related to pillars and groups are related to groups. Each of these terms represents a level of detail and importance to the objective that we shall see in detail later.

This stable *horizontal* relationship offers several advantages:

- Operational priorities can be set because there is in fact a well-developed architectural linkage from left to right;
- New ORs are given context. We can differentiate between genuinely new requirements that further naval warfare (and therefore impact the architectural tiers) and those that improve existing systems (and therefore impact the lower tiers);
- The total number of ORs is reduced from infinite to some finite number because of architectural context. The model helps provide linkage when conceiving and writing ORs; and
- Architectural interfaces—other agencies, other services, other nations—and their technological solutions can be identified more clearly and made more simple.

Vertical Relationships

The elements within a common *vertical tier* of the entire pyramid will be linked programmatically and technologically toward the same architectural subset of the goal. The products on the bottom tier needed to build the subsystem above are related vertically through a core of tiers in between.

This promotes *efficiency*, the second desirable attribute of an ideal acquisition model. From a programmatic and technological sense, a fully detailed Pyramid allows us to see the value of constructing programs that share building blocks rather than building stovepipe systems that tend to multiply building blocks. The simple act of articulating building blocks within existing programs tends to reduce new efforts to build more. This has been true in practice as well as theory: Figure 6 shows the migration of many programs to one in NTCS-A.

Of course, a pyramid is not a triangle; it is three-dimensional. For a particular program, therefore, we should not think of the vertical or horizontal connections through the pyramid as flat on a page, but rather distributed through the volume with bottom tiers providing multiple building blocks leading eventually to a single point on an upper tier. The implication of this is that programmatic tiers can be restructured from today's stovepipe programs toward building block programs that reflect the most efficient means of funding technological families (e.g., computers, receivers, antennas, algorithms, databases) and reduce the overall complexity of systems engineering. This,

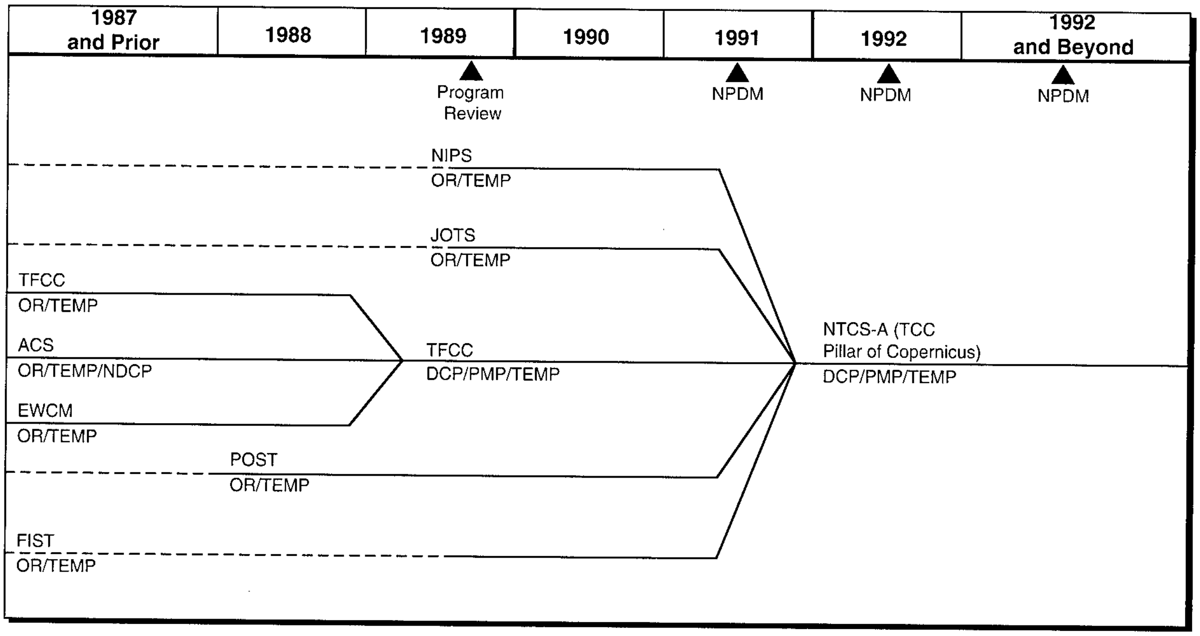


Figure 6. NTCS-A Evolution

coupled with Cyclical Acquisition and the Fleet Assembly Line, provides us the flexibility to *accelerate and decelerate* the production of building blocks—the third attribute of the ideal system.

The stable *vertical* relationships in the model have other important advantages:

- Because building blocks are fewer and more uniform, we can think of each building block as the progenitor of future generations of the same block. They are the genera of future species. In so thinking, we provide ourselves with opportunities to categorize and institutionalize evolutionary acquisition on a larger scale. This will be discussed in greater detail below in the section entitled "The Fleet Assembly Line;"

- Also because of fewer blocks, industry now can see future Navy business opportunities (perhaps several generations before it is needed), and IRAD can be focused to the advantage of both Government and industry. Moreover, small businesses, which in the past two decades have fueled electronics development, can now see opportunities and market to Navy at lower risk;

- Similarly, Navy RDT&E can be more sharply and synergistically focused and very quickly reduced. Moreover, continuity can be established in the use of 6.x RDT&E funds;

- Operational priorities established on tiers above have a directly attributable—and most importantly, measurable, impact on programs and technologies below. For example, accelerating the Electronic Combat Subsystem by two years affects the programs and building block below similarly and quantifiably.

This is in marked contrast to today's structure where programs tend to be an end in themselves;

- Interoperability issues are made apparent at all three Pyramid levels and the number of interfaces—other agency, other service, other nations—is reduced, as is the cost of interoperability to all. Programmatically, the opportunity to capture technologies already funded by other services or in industry is greatly leveraged with a corresponding opportunity for reducing program cost; and

- Finally, the costs of different programs relative to architecture become comparable, and decisionmakers at all levels from action officer to admiral can use the model to develop management tools. Like the arithmetic of refinancing a high-interest mortgage, this appeals to common sense: we should be able to develop and routinely use meaningful lists—the top ten most expensive programs by appropriation, the top ten critical building blocks, the top ten logistics headaches—to cut costs. In the absence of being able to do so, reducing POM dollars is liking trying to cut household budgets by alphabetizing bills and cutting randomly by letter.

Comparison With Past Approach

The Croesus Pyramid model compares with the current approach in three important ways.

Architecturally, our current electronic warfare and command and control systems, which are the predecessors of today's SEW systems, are in reality outgrowths of World War II operational concepts. Platform electronic defense, message broadcasts, networks, and tactical positions dedicated to specific sensors are examples. Not only are many of these systems technologically obsolete, but the operational concepts behind them are divergent and are inappropriate for a new world undergoing technological and political revolutions. By 1989, when SEW was formally established as a warfare mission area, the old operational construct was in advanced decay. Clear architectural goals, present in World War II, were no longer evident. Thus, *the top of the pyramid was missing*, and program elements lacked convergence toward a strategic objective.

Second, as a result of the absence of the top of the pyramid, there could be no clear understanding of what products must be built on the bottom nor could priorities be set among products. Thus, *the bottom of the pyramid lacked foundation* and was too large. Neither technological opportunities nor assembly line techniques could be realized.

Finally, professional focus for operator, resource manager, and engineer alike therefore, tended, by default, to shift away from *operational function to programmatic form*. The result was that the POM and budget process forced programmatic choices, and in the absence of architecture, there was no basis for comparison among dissimilar programs. In OP-094, for example, OBU (consisting of a variety of products from mainframe computers and software to algorithms and storage devices) is weighed against cryptographic hardware programs like the KG-84, which in turn is balanced against communications terminals, whole communications satellite constellations, logistics software upgrades, next-generation computer research, and nearly 400 other individual efforts.

In the absence of the top and bottom of the pyramid, the model can only be a cube—only the center can be discussed. And that today is one symptom we see. The professional dialogue at the FLTCINCs, in OPNAV, and among claimants, laboratories, and systems commanders has become centered around discussions of programs—the center of the pyramid—rather than naval warfare at the top or technological opportunities at the bottom, which have been absent, or at least, obscured. A second symptom is obvious when the various command and control plans of any of the services are opened, they are less plans than descriptions of programs.

Individual programs, therefore, have become almost the sole common denominator today among Congressional, Pentagon, system command, operational, and industry staffs. This fixation on programs has several costly by-products.

- It places emphasis on means instead of ends. It tends to channel and thereby limit operational innovation;

- It also limits technological innovation. Once a program is funded, the inertia is to minimize change to it. When there is a mismatch between programmatic schedules and technological opportunities, schedules usually win out. But this comes around again: shortfalls in procurement dollars lead to over-expenditures in maintenance because obsolete technology was fielded at the outset. This is a chronic problem in electronics; the systemic cause is usually a shortfall in a specific appropriation somewhere along the line;

- It fosters organizational parochialism and self-preservation. Moreover, because of the redundant infrastructure needed to support each program, it is inordinately expensive both in terms of funds and manpower; and

- Fourth, the sheer numbers of programs, and their lack of integration into a common architecture with a strategic objective, make the "learning curve" for all far too long.

Thus, at worst case, *when we focus only on programs, we can only affect change programmatically.* Because programs are too many, too difficult to learn, and too disconnected, the spreadsheet becomes the decisionmaker. In feast, we hurt the taxpayer, but still serve the fleet; in famine, we can serve neither well.

Fleshing Out The Pyramid

Pyramidal Programming reduces these by-products because it eliminates the isolated discussion of programs and replaces it with architectural, programmatic, and technological linkage. The number of tiers possible on the Pyramid, of course, is a matter of professional judgment. We envision eight.

At the top is the warfare area of SEW itself, the strategic objective of the Pyramid. SEW consists of the ability to conduct warfare support and warfare functions. The warfare support functions are achieved through the sequential or simultaneous conduct of four related disciplines: Operational Security, Surveillance, C^4I, and Signals Management. The warfare functions also are achieved through the sequential or simultaneous conduct of four related disciplines: Operational Deception, Counter-Surveillance, Counter-C^4I, and Electronic Combat. See the accompanying box text for a detailed description.

Although all disciplines of SEW will require technology, the three major technological subsystems for Navy to build are the Surveillance, C^4I, and Electronic Combat

Subsystems. The structure of the C4I Subsystem, *The Copernicus Architecture*, was addressed in the first paper of this trilogy.[4]

The SEW pillars (see Figure 7) should be seen at this writing as still incomplete in detail—some are more functionally articulated than others. For example, while TADIXS may be understood both in terms of operations and technology, Electronic Combat technology is still at the platform level, and the doctrine for its force-wide application must still be written. Six architectural pillars currently are envisioned: Surveillance, GLOBIXS, CINC Command Complexes, TADIXS, Tactical Command Centers, and Electronic Combat. However, it seems likely that as we work through the structure of the Surveillance and Electronic Combat Subsystems, additional pillars within those Subsystems will emerge. This is desirable because one can think of pillars as *platforms* —the electronic equivalent of ships, airplanes, and submarines. As we shall see, the idea of *electronic platforms* will allow us to approach design and installation of SEW systems in a modular fashion.

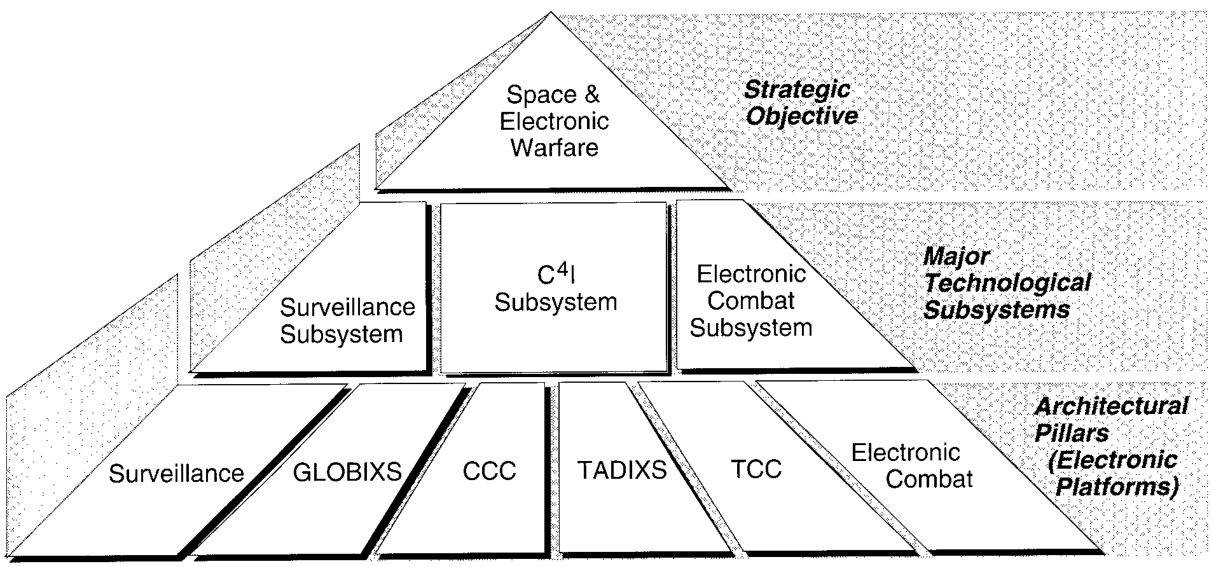


Figure 7. Upper Tiers Of The Pyramid

---

4 The most comprehensive work to date is the *Copernicus Requirements Document*, OP-094, April 1992.

SEW DISCIPLINES

SEW includes both warfare and warfare support functions, contained within eight disciplines. Warfare support disciplines are Operation Security, Surveillance, C^4I, and Signals Management.

Operational Security consists of measures taken to minimize hostile knowledge of ongoing and planned military operations. Operational Security includes physical security, counterespionage, and personnel security.

Surveillance includes the tactical management of all technical surveillance as a force system across the air, land, sea battle space, including all sensors, regardless of location (whether national, theater, or platform) or ownership (whether service, joint, or combined.)

C^4I is the means to the command and control ends. C^4I is a technological, organizational, and doctrinal system that provides three functions: the doctrinal delegation of forces (i.e., command and control); information management and display (i.e., communications and computers); and intelligence (i.e., estimation of capabilities and intentions.) Since World War II, in modern warfare, the function of command and control has been facilitated through the system of C^4I.

Signals Management includes measures to protect force signals, including frequency management, signals security, communications security, computer security, transmission security, and emission control management.

The warfare disciplines of SEW are Operational Deception, Counter-surveillance, Counter-C^4I; and, Electronic Combat.

Operational Deception incorporates more than electronic deception. On the modern battlefield, Operational Deception begins with diplomatic posturing, ends with technical reinforcement, and includes a multiplicity of actions in between. Operational Deception occurs in two phases, preparation and execution, and it is intended to influence enemy plans, execute a stratagem, induce reactions over a short period, and apply pressure to act. Operational Deception techniques are conditioning, reinforcement, and continuity across echelons and components. Operational Deception is an essential element of every military action, and multi-echelon, multi-component coordinated Operational Deception is necessary in combined arms actions.

Counter-Surveillance targets the enemy's surveillance systems. It is the sum of all active and passive measures to prevent enemy surveillance of areas occupied by own forces. It consists of techniques to deny detection, divert detection, deceive or overwhelm the detector, and destroy it. In modern warfare, Counter-Surveillance is accomplished at all echelons, from unified commander through joint task force commander to component commander.

Counter-C^4I targets the enemy's C^4I systems. It includes measures to deceive, delay, degrade, or destroy elements of a hostile C^4I system, including his communications, data, and command and control nodes. It consists of techniques to deceive, saturate, jam, and destroy such elements. Like Counter-Surveillance, in modern warfare Counter-C^4I is accomplished at all echelons.

Electronic Combat targets the enemy's weapons and weapons systems. It includes the coordination of all measures to provide counter-targeting and counterweapon/terminal phase protection to the force. The aim of Electronic Combat is to protect the force by providing a doctrinally organized, technologically seamless area defense. However, unlike point electronic defense of today, Electronic Combat will accomplish that force defense both through actions traditionally viewed as offensive (e.g., destruction of enemy radars) and defensive (e.g., classical electronic counter-countermeasures)—the best defense is often offense.

Architectural Groups

The strength of the Pyramid depends both on the clarity of horizontal and vertical linkages and the granularity of detail in the tiers. For that reason, while all tiers are important, the tier that transitions from pillars to programmatic elements is especially critical. That tier we call "Architectural Groups." Architectural Groups can be best understood as a collection of program elements that share a common architectural goal. They define *the number of distinct assembly lines* needed to build the electronic platforms above.

Organizationally, both within OP-094 and SPAWAR, Architectural Groups cross division boundaries, fostering both cooperation and competition among technological and programmatic approaches within the group. Both the number of Architectural Groups and the number of Program Elements below them are the subject of internal staffing within Navy, at this writing. This is as it should be for two reasons:

- Funding changes require cooperation and approval from several layers of authority; we will implement *The Croesus Strategies* in concert with these authorities; and

- As illustrated in Figure 8, which shows the percentage of Total Operational Availability (TOA) by several existing categories, both the categories themselves and the percentage of TOA assigned to them have been consistent for at least a decade. This is illogical in the face of new architecture and new building block technology. Today, these categories no longer reflect what we need to build, nor do the percentages of TOA assigned to them reflect the amount or appropriation necessary to build them in the 1990s.

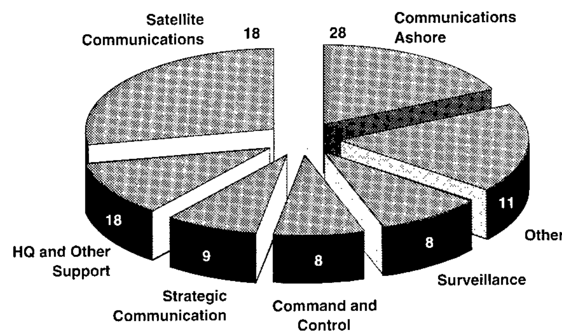


Figure 8. OP-094 TOA By Categories

In a world in which information systems technology is proceeding apace at 18 months per generation (see Figure 9), the percentages of TOA by the categories in Figure 8 are completely out of proportion with industry developments. They represent missed opportunities so long as we continue to use them.

One can also think of Architectural Groups as new categories linked to a new architecture: each of the pillars in the tier above requires one

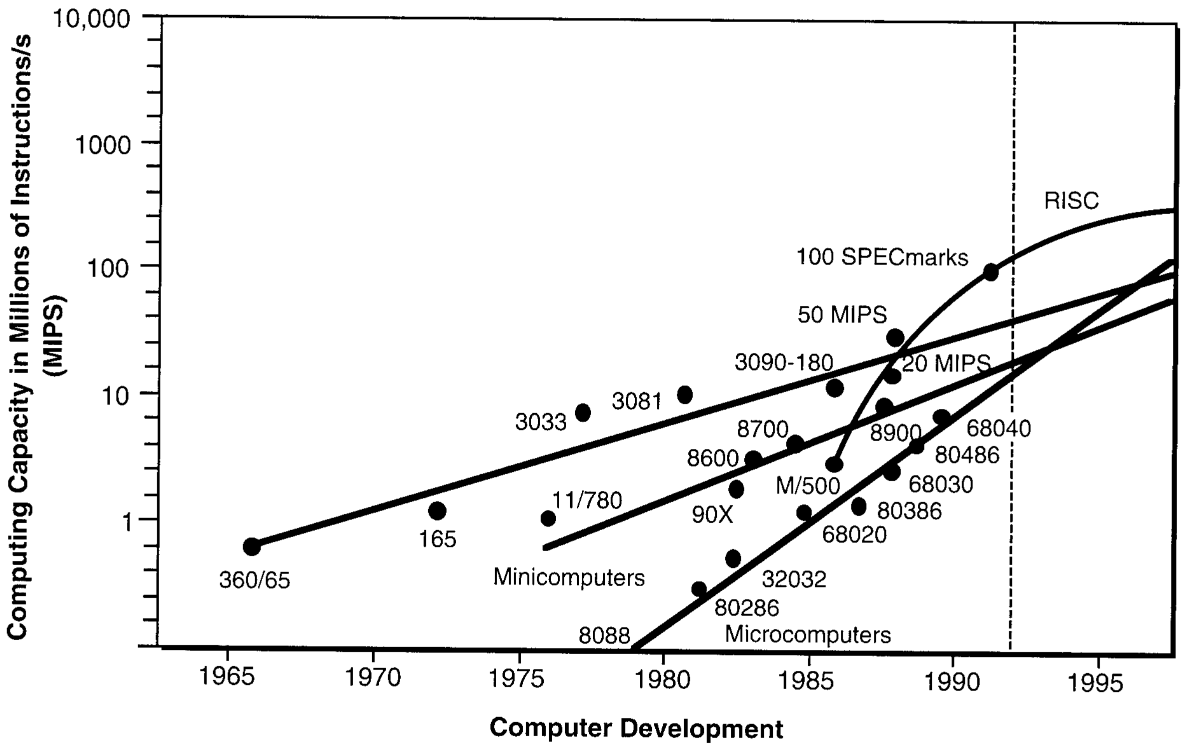


Figure 9. Development of Computer Technology Since 1980

or more Architectural Groups below. The number of Groups necessarily will change over time as a function of technological and architectural change; however, the minimum stability for a group probably should be set at three POM cycles (6 years.) At this writing, a reasonable number of Groups seems to be 10-15, with 30-40 Program Elements in the tier beneath. While we are some distance from a strawman on new Program Elements, Figure 10 shows a Group structure currently under consideration.

Architectural Groups are a good managerial focal point of the Pyramid. It is here architecture, programs, and technology converge with a level of detail that does not require specialized expertise. Because of that, funding tradeoffs can be made for the first time against architectural priorities; it is in this tier where critical paths in operations, programming and technology converge and become most obvious to managers. At the Architectural Group tier, the Croesus Pyramid makes visible several opportunities:

- Groups provide the logical basis to restructure existing programs around architectural pillars, to prioritize among requirements, and to accept budgetary cuts while retaining overall group goals and momentum. Thus, today's long learning curve is reduced, and the spreadsheet made less tyrannical because its impact is more visible and, therefore, more studied decisions can be made.

- The Pyramid also allows us to press into Government service industrial assembly line techniques. The metaphor of an

SEW Disciplines	Technological Subsystems	Architectural Groups
Warfare Support • Operational Security • Surveillance • C4I • Signals Management **Warfare** • Operational Deception • Counter-Surveillance • Counter-C4I • Electronic Combat	Surveillance C4I Electronic Combat	• Surveillance • GLOBIXS • CCC • TADIXS • TCC • Electronic Combat • SATCOM Infrastructure • Common Equipments • General Support • RDT&E Support • ILS Support • MPT Support • Architectural Support

Figure 10. Architectural Groups

automobile assembly line is apt. In the absence of the top of the pyramid—a desired "product"—the entire factory is unmanageable; *there can be no assembly line, only assembly.* Groups are where assembly line goals are best set.

- Groups help us use appropriations better. We can see the impact of cuts in RDT&E or O&M,N, for example, by using the Pyramid. Groups provide us with the opportunity, when coupled with Cyclical Acquisition, to conserve all funding appropriations as the assembly lines proceed toward product roll-out. In automobile manufacturing terms, if we can reduce the cost of raw materials or the cost of running the assembly line, we can produce more cars for less capital outlay. In Navy terms, we may be able to put greater quantities of better equipment in the fleet in leaner years ahead than in fatter years preceding.

These savings can be achieved at this tier because Architectural Groups give us the means to do thread analyses horizontally across the pyramid at the vertical point where architecture and programs meet. See accompanying box text.

When we consider such savings, we illustrate our point: better technology can be put into the fleet faster in the next decade with far less dollars than the last. The Croesus Pyramid helps us see opportunities like these.

It is for this reason that the Copernicus Architecture was not funded as a program per se. Its purpose is to provide the upper tiers of the pyramid so that Program Elements might be restructured and realigned toward its goals.

CASE STUDY 1: BETTER COMPUTERS CHEAPER

A case in point to illustrate thread analysis is instructive. Figure 11, a horizontal cut across the pyramid, shows a series of five computers currently in use in the delivery of shore-based sensor data to a tactical platform at sea. They are positioned in operational sequence, and their instructions per second, a measure of computing capability, is compared.[5]

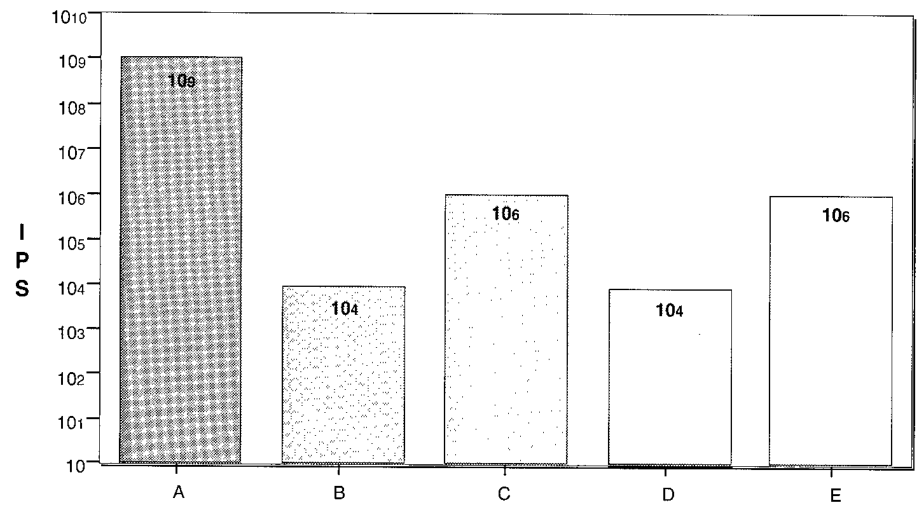


Figure 11. Sensor-to-Shooter Computers Compared By IPS

Computer A, which processes data from the sensor, operates at billions of instructions per second (BIPS). Computer B, which is a Special Compartmented Intelligence (SCI) communications processor, is capable only of one thousand instructions per second (THIPS). Computer C is used to fuse the product of A and other sensors delivered by B; it is capable of one million instructions per second (MIPS). Computer D sends the fused messages to sea; it, like Computer B, is a THIP machine. Finally, display of the information in an afloat tactical command center is done by Computer E, a 23-MIPS machine.

This is the engineer's perspective, and from his perspective, nothing seems wrong. BIPS are appropriate for computer functions required in processing some raw sensor data; a THIP machine is suitable in terms of hardware for message processing.

---

5 An instruction is a command to the computer. The term usually refers to machine language instructions that only the computer understands. Machine instructions are made up of two parts: the operations code and the operands. The operations code specifies the type of instruction or action to be taken (e.g., input, add). The operands are the references to data or peripheral devices. Instructions in the aggregate make up software. Small sets of instructions are called subroutines, program modules, or functions. The term instruction per second or IPS is used as a measure of the processing capability of a computer. Although increasingly IPS are being abandoned in favor of a newly developed and more accurate measure of comparison called the SPECMARK, IPS remain a reasonably accurate means to compare computers developed in the 1980s. As a point of comparison, a personal computer using the INTEL 486 chip is capable of about 5 million instructions per second.

Figure 12 shows the same sequence from an operator's point of view, comparing maximum time delays in output experienced during Desert Shield/Storm. Clearly visible—from an operator's perspective—the communications centers (perhaps the computers, perhaps some procedure, or perhaps the software) represent the critical failure.

Let us change perspectives again, this time to the resource manager. Figure 13 shows the computers again, this time by cost. (It is important to note that this is a minimum estimate, since it only encompasses purchase cost, not operations or maintenance cost—a point to which we will return to later.)

If we look across the pyramid from each perspective, then, we can begin to look for the classical cost benefits of spending the least funds for the best return. In fact such an analysis was done in OP-094, and Computers B through E are all being replaced by the TAC-3 computer. Moreover, the communications processors will use common software.

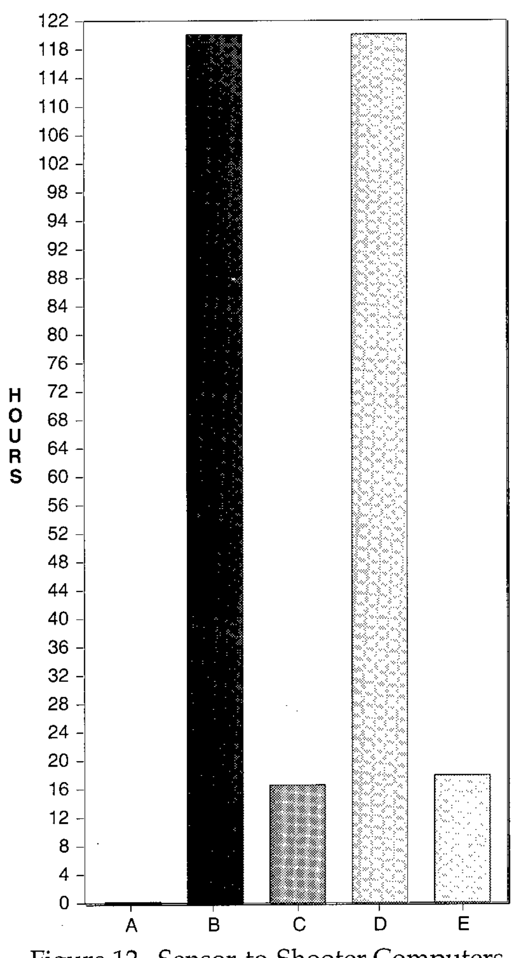


Figure 12. Sensor-to-Shooter Computers Compared By Throughput

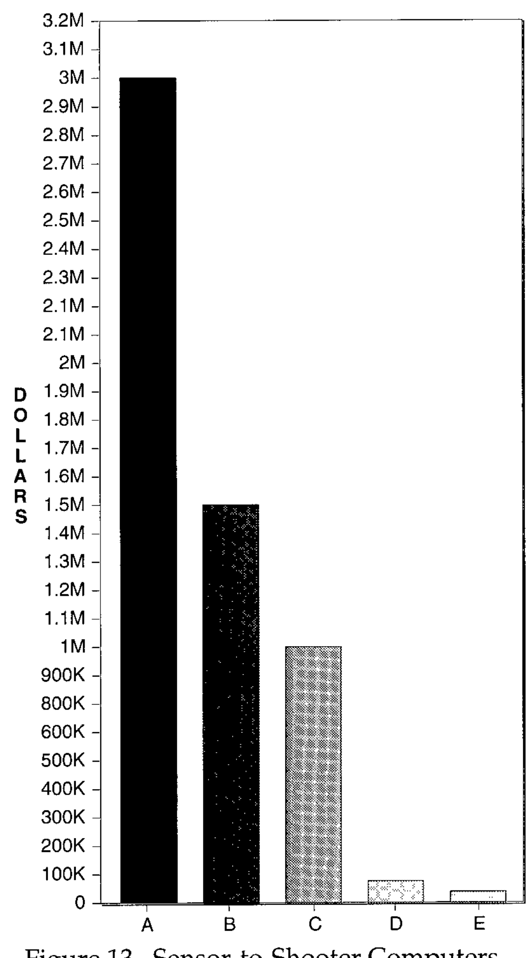


Figure 13. Sensor-to-Shooter Computers By Cost

The Middle And Lower Tiers: Breaking Stovepipes

If the purpose of a programmatic structure is to relate means to ends, and both have changed, then the programmatic structure also must change. Simply moving existing Programs Elements into new Architectural Groups does not make flawed programs better. Once we have constructed the upper tiers of the Pyramid, however, we can—through thread analysis and a process described below called "best of breed"—produce fewer and more suitable programs.

The fewer the building blocks on the lowest tiers, the better from all standpoints: operations, interoperability with other services, maintainability, training, software and hardware refreshment are all facilitated.

However, we can achieve fewer building blocks only by having an architecture, which SEW in the form of the Pyramid provides, and by conforming to industry standards wherever possible. The rewards of doing so are demonstrable. Figure 14 shows as an example the number of tactical data processors in use by Navy three years ago. Today, these are being replaced by TAC-3-series computers and a common operating environment.

Once we are able to describe the architectural building blocks functionally (e.g., standard communications processors and network software; standard storage devices and data base software) within a common operating environment, a step-by-step approach can be taken to move beyond today's many vertical stovepipe systems. This move involves five steps to be repeated over time on

System	Processing Hardware	Processing Software	Display Hardware	Operating System	Dbms	L.O.C.	System Name
ASWOC Upgrade	DTC-2 Sun 4/110 & 330	Ada, C	Sun	UNIX	ORACLE	700K	ASWOC Modernization Program
CMST-N	Sun 4/370	C, FORTRAN	Sun	SUN OS 4.1	SYBASE	100K	Collection Management Support Tool-Navy
ENWGS	HONEYWELL DPS-8	PL 1	BARCO & Sun Terminals	MULTICS	EMBEDDED (MRDS)	450K	Enhanced Naval Warfare Gaming System
FPC	Macintosh, DTC-2, * (In transition)	C, FORTRAN, LISP, Ada	DTC-2	VMS, UNIX, DOS	ORACLE	>500K	Fleet Planning Center
FHLT	DTC-2 Sun 4/110 & 330 (In transition)	Ada, C	DTC-2 (Dual Monitor)	UNIX	ORACLE	>750K	Force High Level Terminal
JOTS I	HP 9020 A, C	BASIC	HP 9020 A, C	ROCKY MTN. BASIC	EMBEDDED	250K	Joint Operational Tactical System
JOTS II	DTC-2 Sun 4/110 & 330	Ada, C	DTC-2	UNIX	SYBASE	330K	Joint Operational Tactical System
NWSS	HONEYWELL H-6000 or DPS-8	COBOL	TEK 4014, WANG	GCOS-8	Integrated Data Store (IDS)	1,300K	Navy WWMCCS S/W Standardization
OBU	VAX 8650 MICROVAX II	PASCAL, FORTRAN	VMI Alph/Numeric GENISCO Graphic (to be replaced)	VMS Ver 5.2	IMBEDDED & ORACLE	975K	OSIS Baseline Upgrade
OSS	DTC-2 Sun 4/110 & 4/330	Ada, C	DTC-2 BARCO 1001	UNIX	ORACLE	>500K	Operations Support System
STT	DTC-2 Sun 4/110 & 330	Ada, C	DTC-2	UNIX	ORACLE		Shore Targeting Terminal

* Collection of prototypes transitioning to indicated systems.

Figure 14. Table of Tactical Decision Aids

existing programs, where necessary, during the next several years. On completion, the bottom tiers of the Pyramid will be in place, and its foundation built from modern technology.

- The first step is to develop detailed functional engineering models to provide an engineering template from end-to-end.

- The second step is to devolve existing programs into potential building blocks and select the "best of breed" among the blocks suitable for use in the architecture. This will necessarily be a "cut-and-paste" task, with the number and diversity of building blocks varying by program. Some programs (e.g., Navy Standard Teletype, KG-84, Combination Radio) may already be building blocks. For comparative purposes, this process will slice existing stovepipes into components.

Engineering criteria of suitability, feasibility, and affordability will be established. Affordability (which heretofore has been a programmatic consideration rather than an engineering consideration per se) is a legitimate criterion in this step because, when applied to building blocks rather than whole programs, cost savings is more easily quantified. Cutting raw material cost gains more faster than rebuilding the whole factory.

Indeed, this is one of the great benefits of horizontal architectures over stovepipe programs. Stovepipes only can be compared against other stovepipes that typically are not being developed to meet similar requirements. Affordability today can only be a POM issue rather than a building block issue because, in the absence of direct comparison by function and requirement, there is only the question of whether enough money remains in the POM at the end of the process for all programs (e.g., TACINTEL vs. Mini-DAMA, or JTIDS vs. OBU).

In a "horizontal" comparison, however, communications processor "A" from program "A" can be compared with processors "B" through "Z" from other programs competing to select a Navy standard communications processor.

Families of building blocks will arise: there will be two kinds of communications processors in Copernicus—an ashore processor and an afloat processor. Both types will come in several versions; for example, the shore processor probably will be implemented as a circuit card, in a workstation, or as a stand-alone machine depending on size of node and other considerations. Figure 15 illustrates "best of breed" selection.

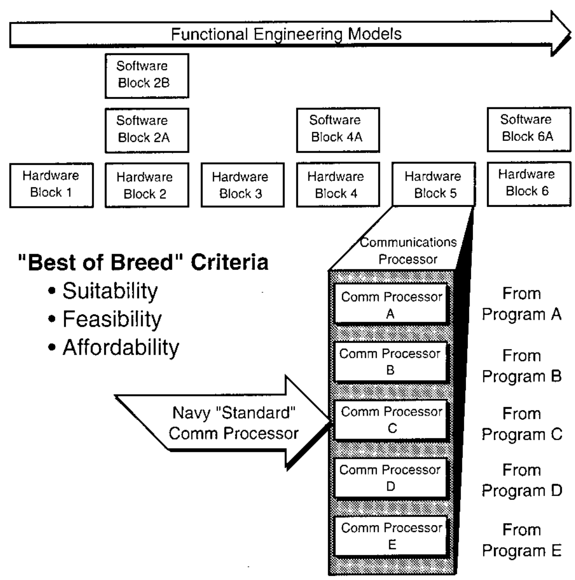


Figure 15. Best Of Breed Analysis

- The third step is to overlay the resultant existing best-of-breed building blocks against the desired functional model. Building blocks that could not be identified in a best-of-breed competition are candidates then for new programs. Similarly, if a best-of-breed winner does not fully meet the functional requirement of the building block, we may either improve it incrementally or order the development of a new prototype. Figure 16 illustrates this process.

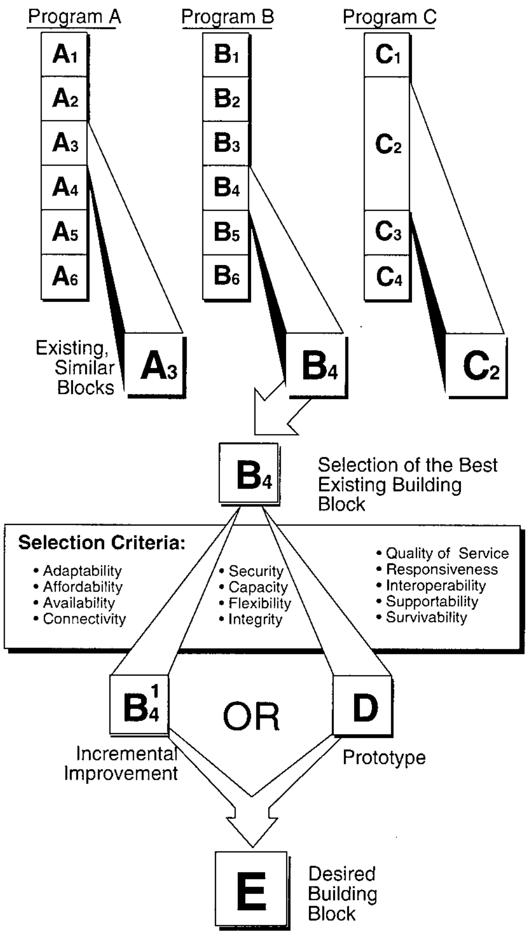


Figure 16. Breaking Up Programs

- The fourth step is to develop an ILS strategy for each major block family. This is an important and money-saving step that will be discussed below under the Fleet Assembly Line.

- The final step in the process will be to restructure programs that require it.

This will be a complex process occurring over several POMs in which three types of programs will emerge: building block and RDT&E programs, which provide the basis for the third type, pillar programs. Two of the three types of programs will contain several appropriations. For example, the eventual implementation of the Norfolk CCC will evolve from a restructured OSS program that will draw resources from several building block lines. The Norfolk CCC will require OPN and O&M,N, but little RDT&E. Building block programs (e.g., TAC-3) will require RDT&E, OPN, and O&M,N.

Implications

Pyramidal Programming has four revolutionary implications.

First, at the top level of the Pyramid, architectural pillars (e.g., GLOBIXS, TCCs) are really *platforms*—the electronic equivalent of ships, airplanes, and submarines. The idea of Program Elements themselves sooner or later will give way to the notion of pillar-platforms. These electronic platforms have clearly definable goals, production quotas, interfaces, and composition.

Second, the articulation of electronic platforms clears the way for *modular installations* on a grand scale. New production ships or ships in overhaul need only provide space, power, and fiber optics into which will be placed an electronic platform with the proper form, fit and function. In a Navy in which our current ships and aircraft will be with us in 2010, modular electronics is the means to keep them at the cutting edge.

Third, there will be a cultural shift in acquisition from a mentality of *builder* to *shopper*.

Although there will continue to be Government-unique building blocks, increasingly those building blocks will also become progenitors and be used over and over. The remainder will be commercial.

Fourth, as we transition from stovepipe programs to building block programs, we will require programmatic flexibility. Buying new technology will mean finding a way to mine the ore we have rather than look for new veins. In turn, this will require a shift in the way we conceive programs. A significant savings can be gained through modernization and new management approaches. Telecommunications leasing, which represents a significant portion of TOA, is a case in point. Improving Navy shore telecommunications will mean migrating stovepipe lease programs to commercial rate structures. Unless there is a flexibility to make that migration, there will be no incentive to modernize. Both taxpayer and operator—who are one in the same—will lose.

CYCLICAL ACQUISITION

SEW arises from our understanding of the importance of electronics and computer systems and of the importance of exploiting hostile SEW systems-computing, information, electronic combat technology--on the modern battlefield. Yet, when we consider today's process for acquiring SEW systems, we find ourselves on the horns of a dilemma.

Despite our understanding of their importance, we find that off the battlefield and in the market place technological opportunities often are difficult for Government to exploit. This is due to two factors.

- First, our current acquisition system is maximized for large programs that yield products that are typically stable in design and designed for more than a decade of use. Neither is characteristic of electronics and computer systems.

- Second, there is a widening gap between the speed of the acquisition process and the much greater speed today of technological change in computing and electronics. This is becoming more and more obvious to all. And, not surprisingly, from all levels of DoD and from industry, there is increasing concern and impetus to find a new construct for electronics and computer acquisitions.

Such a construct is decidedly possible today for Navy SEW systems—and potentially for other technologies with short generations lengths as well. Moreover, it is practicable today, while still remaining consistent with the existing acquisition policy of the Secretary of the Navy and the Department of Defense.

An Acquisition Engine

Pyramidal Programming only identifies building blocks; it does not build them. And since the building blocks include those built by industry, those built by other Services, and those built uniquely for Navy, in the future we will have to shift our thinking as to what acquisition *is*. This shift in thinking, as we saw above, is one from building to shopping. It will have both organizational and manpower implications—as well as financial rewards. It also will change the *process* of acquisition.

Think of acquisition as an engine. The input is money. The output is technological building blocks. The engine's purpose is to produce output from input as efficiently as possible. There are four sets of effectiveness measures: effectiveness, suitability, affordability, and sustainability.

Effectiveness has to do with value at sea: does the building block work well, can it be improved, is it interoperable? *Suitability* is a technological issue. Measures of effectiveness (MOE) might include maturity, functionality, and modularity. *Affordability* addresses such concerns as percent of TOA applied to the block, procurement strategy, and economies of scale. *Sustainability* involves maintenance, provisioning, installation, and training.

As we work with these four sets of MOE, we find the boundaries between them start to pale. Navies by definition *are* technology; that is why war at sea has long been quick, violent, decisive, capital intensive and rare. Technology depends on money: that is one reason why historically few nations have become sea powers. Sustainability should depend on a specific technology not a monolithic ILS philosophy: surely installing and repairing an antenna is different than installing a computer. Similarly, new technology improves sustainability and lowers costs: the development of plug-in telephones saved telephone companies millions in installations and manpower costs.

We can draw three important conclusions from this:

- The ideal acquisition engine for electronics and computing technologies must be cyclical to capture the interrelationships of all four MOEs.
- The engine, therefore, can and should be fueled by all four MOEs. A system that inputs only ORs is too inefficient and misses opportunities, both technological and programmatic; and
- Such an engine, if made to measure and conserve costs throughout the cycle, can put lower cost technology in the Fleet faster (because we shopped for it) that costs less to support (because its mean time between failure is longer than its mean time before obsolescence.) A dollar saved, whether RDT&E, OPN, or O&M,N is still a dollar earned.

Our Current System

Unlike our engine, the current system can be divided in practice (if not precisely by instruction) into four separate processes: the development of ORs; the development of the POM; the acquisition and procurement of systems; and the support of systems, which includes installation, training, maintenance, and provisioning.

These four processes, while originally intended to be iterative, have become linear and are managed in isolation from each other. As they become more linear and less iterative, *the entire cost of a program rises higher and yet becomes less visible.* The total amount of RDT&E, OPN, FPN, O&M,N, MPN and other funds used to build, buy, install, operate, and maintain a system or a group of systems often simply never is calculated. From a budgetary standpoint, this strategy is wasteful.

There are also operational disadvantages to linear acquisition. Although they vary in quality, ORs often suffer from systemic problems. We have seen asserted that ORs can fail to capture tactical opportunities presented by technological advances. They are also difficult to coordinate among FLTCINCs. If they spawn a program, it rarely looks like any single FLTCINC's requirements. Accommodation of joint interfaces and appreciation of allied interoperability similarly is not easy. Existing programs, on the other hand, because they are funded and so hard to get started, take on a concreteness that requirements do not. Because of this, we are in danger of having the development of naval technology evolve from *a process to further naval warfare to one that furthers specific programs.*

The POM process magnifies and exacerbates the irrelevance of bad ORs. System commands, saddled with a POM that usually doesn't match the CINC requirements, burdened by paperwork intended to bulwark and formalize the process, and cognizant of the tendency toward technical shortfalls in ORs, tend to turn to preconceived technological solutions and to make the system work, molding it accordingly. Once the equipment has been procured, the process suppresses innovation.

Support for systems is also problematic. When a system becomes difficult to maintain or operate—or expensive to install or train personnel—it is rare the problem surfaces, rarer still a timely requirement is generated to replace the system, and rarest a system is restructured to capture technological opportunities. There are at least two reasons for this.

In practice, linear acquisition directly responds to an OR, budgets for it, acquires it, fields it, and typically leaves it stranded. Life cycle support is too often simply life support. This has been especially true of electronics and computing systems in the last 30 years, during which systems engineers built end-to-end stovepipe systems from a large menu of what was, in hindsight, unstable technology. These systems required diverse hardware, often used proprietary or other unique software, lacked standards, and required high training and maintenance costs. Interoperability seemed a labyrinth. Until the advent of open-systems standards and the general migration of computers to a few families of languages, operating systems, and microprocessors, technology refreshment to recoup support costs simply was not practicable--you had to build a replacement stovepipe.

Second, there is often a disincentive operating. When an organization decides to spend a percentage of limited RDT&E and OPN to buy a new system, it usually does so in response to one of many ORs it has received, which is the starting point of the linear system. If that organization is not also responsible for the installation, logistics, and support of the system, it rarely seeks statistics on those costs since requirements do not directly impact on that organization's position in the system. If statistics were obtained, the funds saved by fielding a new stovepipe system would not be recouped by the same organization that paid for the new system. *Therefore, from an organizational view, there is no incentive to modernize.* In the 1980, when old stovepipes required new stovepipes, this mentality was understandable, if not laudable. In 1990s, open-systems standards and architecture make functional replacements cost effective.

Fueling The Engine

The *strategy* behind Cyclic Acquisition is to converge the four divergent processes above into a cycle and to monitor and conserve all appropriations at critical junctures within the cycle. Those junctures reflect the areas in electronics acquisition that typically pose opportunities to reduce funding or insert more advanced technology.

By converging linear acquisition into a cycle, the new process can accommodate formal points of entry other than the OR. If the points of entry coincide with the areas in which statistics can be gathered, operational, programmatic, and technological opportunities can be captured.

Electronics and computing technology has four such junctures as shown in Figure 17. If we structure and formalize those junctures, we see a set of four kinds of requirements surface.

- First, a change in threat or doctrine will continue to produce new or changed ORs;

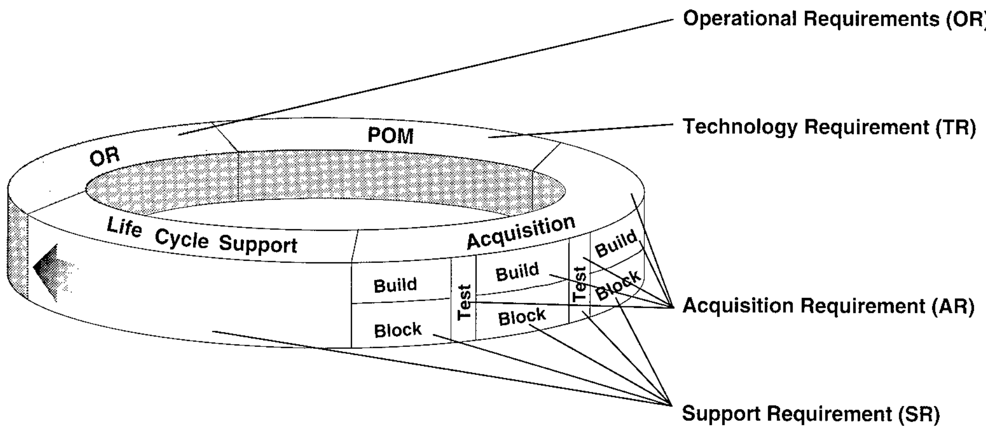


Figure 17. Cyclical Acquisition

- Second, advances in technology will present opportunities that in writing we may call "Technology Requirements" (TRs);

- Third, programmatic cuts or additional funding will drive time frames for fielding equipment. A cut may make an OR or a TR no longer feasible. Documents that reflect such changes would be called "Acquisition Requirements" (ARs); and

- Fourth, as a system grows older, or as a new technology becomes cheaper to support, "Support Requirements" (SRs) would be generated.

These requirements can form the basis of an electronics and computer acquisition instruction that, while remaining consistent with the DoD 5000 series, institutionalizes evolutionary acquisition, paving the way for Navy to come into the information age.

As with the Case Study involving computing above, we have used these four junctures in practice today in pilot programs with success. The accompanying box text details a second Case Study involving the restructuring of a whole program by stepwise moving through the four MOEs.

Cyclical Acquisition builds on Pyramidal Programming and fuels the Fleet Assembly Line. Pyramidal Programming conveys three advantages. It relates architecture, programming, and technology in a common model that makes the consequences of change visible to decisionmakers. By relating programs to architecture, it reduces stovepipes, which in turn reduce the sheer number of products produced on the bottom tier. Finally, it facilitates thread analysis and other management tools that surface problem areas faster.

Cyclical Acquisition adds to that the incentives to conserve all appropriations, thereby making money available for new technology as well as real savings returned to the taxpayers in terms of reduced budgets. It captures the dynamics between technology and operations through multiple entry points into the cycle instead of the single OR. It institutionalizes innovation in the form of TRs so that technological refreshment and prototyping can be realized.

The Fleet Assembly Line, to which we will now turn, provides the means to put the products into the Fleet.

CASE STUDY 2: REPAIRING WHOLE PROGRAMS

Our traditional orientation is so ingrained that at first there may seem little difference between linear and cyclical acquisition. A closer view, a case study, is therefore, useful.

In 1986, a TOR was written for improvement of an existing communications network used for exchanging a particular kind of sensor data. The TOR called for multi-frequency satellite communications data network (2400-4800 bps) fully nine years before the scheduled implementation of EHF in Milstar and the Fleet EHF Program and at a time when U.S. Navy HF equipment could provide a throughput of only 600 baud. At the same time, the Navy UHF constellation was maximized in terms of throughput and the UHF Follow-On constellation to replace it was slated to provide no additional capacity.

Although the TOR connected multi-service components ashore, it provided no joint interfaces at the tactical level. Its purpose was to exchange uniquely formatted messages among systems ashore and afloat. When the program was funded, its implementation approach was to develop system-unique communications software, to operate on what would be a 30-year-old computer, at the estimated IOC date of 2003.

By 1989, all appropriations in the program had been cut except RDT&E, which (still aimed at producing the system above) was in excess of $130 million over the SYDP and was being used to implement other programs. The program, which could not be engineered as the TOR required and lacked appropriations to field it if it could, was going nowhere—despite a bona fide operational need to improve net size and throughput in the existing network and its position repeatedly on the CINC's Integrated Priority List. By Desert Storm, five years after TOR, Navy was forced to install an additional net of the existing configuration in the Indian Ocean to accommodate the number of subscriber ships that needed to copy it.

At that point new approaches were sought, and the methodology used was to invent Cyclical Acquisition with each of the four processes studied and merged.

- *Operational Requirement.* The first step was to review all related ORs with an eye toward implementing them in a unified coherent architecture. There were three unfunded TORs, another program at Milestone IV, and the program above. These were reexamined and merged by cataloging the information requirements in the networks and by transitioning those requirements to common formatted digital products. In the place of many uniquely formatted messages, two digital products were substituted. Once the information was grouped and encapsulated in digital format, an analysis of the originators and potential users of that information, including joint and combined users, was undertaken, and other criteria such as timeliness of data flow was established. The resulting architecture was reviewed, not from a communications perspective, but from an *operational perspective* to determine the impact such technology would have. Finally, the applicable joint requirements were adopted for resulting operational positions.

- *Technology Requirement.* The second step was to reduce the number of technological components to the least possible and to standardize those that remained. This was achieved by building similar network software for all the networks required, eliminating the need for four distinct network development programs. A similar decision was made to host them on a single computing engine, the TAC-3. The TAC-3 replaced four computers. A decision, part philosophical (e.g., communications programs buy capacity) and part practical (i.e., capacity bought by other programs might slip programmatically), was made to build a UHF multiplexer into the resulting consolidated program to provide additional capacity for the networks. Finally, since information was the desirable commodity, not data, operational man-machine interfaces were devised that presented data from multiple sensors on a common display. This will make use of another software family—JOTS II. The networks were constructed so that any operator on one of the networks afloat could be connected to any operator on one of the networks ashore, exchanging the common digital products.

- *Acquisition Requirement.* The consolidated program would result in the fielding two operational positions, each consisting of a TAC-3 computer, one UHF multiplexer, and five software layers. Because the program was structured to capitalize on building blocks from other existing programs, its cost was reduced by $30 million over the original five program estimations. IOC was reduced by 10 years; FOC by five years.

- *Support Requirement.* At the first developmental stages of the programs, those organizations selected to do software maintenance were involved. All hardware blocks of the program are duplicated in other programs, and most of the software will be duplicated. ILS approaches, therefore, will be consistent with those other programs. Embedded training was built into the positions. The dedicated communicator billet will be eliminated.

THE FLEET ASSEMBLY LINE

The third Croesus strategy is focused on two opportunities:

- Changing the pace of technology injection into the Fleet in response to operational tempos and generational changes in technology rather than individual program funding; and

- Conserving funding through "Just In Time" (JIT) assembly.

When the carrier USS Washington was commissioned she was moved to another pier the following day to remove the NAVMACS communications processor from her that was obsolete on installation. A modular electronics approach has not been planned. Similarly, the Trident submarine USS Kentucky recently steamed from her ways with KW-7 cryptographic machines installed, more than a year after active ships in the Fleet removed them from their radio rooms.

These are signs of hemorrhaging in a system which was designed to deliver technology to the Fleet in response to a different threat, with different budgets, and in a different technological era. When they are coupled with the high cost of linear acquisition and then couched in manufacturing terms, the results are clear. The product specification can be vague. Its relationship to other products being manufactured may not be clear. Other similar products or major components of the product may already have been built elsewhere. Cost considerations may not be visible along the entire assembly line. Manufacturing costs may be a guess. They may reflect the capital of the firm rather than the cost of making the product. Logistics costs and installation costs may be similar guesses. Product performance and obsolescence may not be monitored. Finally, customer feedback is often ignored.

It is appropriate, then, to conclude this paper with a proposal to improve support to the customer—to redesign the assembly line. To do so, we return briefly to the Pyramid and specifically to Architectural Groups, where sandwiched between architecture and programs, the assembly line is most obvious.

Groups

Architectural Groups are representative collections of programs designed to achieve a common goal. Programs within a Group aimed at building *GLOBIXS* have different engineering and testing milestones, production quotas, and fielding targets than another Group aimed at building the *TCCs*. Because Architectural Groups represent the midpoint transition from building blocks to system objective, and because different Groups have different characteristics, they also represent the number of assembly lines that are needed to build SEW.

The GLOBIXS Group provides a good example. We plan to build eight GLOBIXS. Each of the GLOBIXS will use the same facilities: a common workstation terminal, modems, and cryptography, for example. Nearly all software will be common except for the topmost application layer that defines the GLOBIXS' purpose (i.e., ASW, Power Projection.) Like a row of identical office buildings housing different corporations, technologically, each GLOBIXS will be constructed in the same way.

At the Group level then, decisionmakers need only decide which GLOBIXS have construction priority and set timetables. At the tiers below, funding will be apportioned from the total TOA, and building block technology will mirror those priorities—or if they do not, that discrepancy will be visible at the Group level. Program funding levels for building-block programs can then be set to reflect the number of building blocks needed to meet the assembly line timetable.

Two factors should influence the speed of the assembly line: the priorities within Groups (e.g., ASW GLOBIXS versus Power Projection GLOBIXS) and the operational tempo of the Navy components or Fleet units, which are the customers. This is in marked contrast to today, where the assembly line is driven by the funding levels of individual programs.

It also sets the stage for a classical example of where Total Quality Management JIT techniques conserve dollars. Our current system is like an assembly line in which doors, transmissions, engines, and bodies are bought independently, based on the budget of individual organizations, and then stacked on the line without respect to their requirement on the line.

In Navy terms, buying 500 radios in a year when only 200 can be installed usurps funds for other raw materials urgently needed in the line that year. This is a somewhat simplistic view—there are, of course, economies of scale to be considered. However, economies of scale versus costs of storage, obsolescence in electronics, and other considerations are calculable--and, more to the point, should be calculated.

When we create the assembly line in this manner, we conserve dollars because we bring to the line only the raw materials needed when they are needed. When we build those raw materials to bring to the line through Cyclical Acquisition, we ensure the cost of the raw materials is the lowest possible and foster their continued improvement through the four requirements points in the cycle. There are three additional advantages beyond cost savings and technology refreshment.

First, in manufacturing terms, the assembly line can be sped up or slowed down in response to the customer. In Navy terms, as the operational tempo is increased, equipment can be inserted into deploying battle groups as they depart. This in fact was done during Desert Storm, but only at great cost and significant disruption because programs had been paced to reflect their individual funding levels without overall operational context. Illustrations are manifest: commercial electronics equipment from laptop computers to antennas were sped down industrial assembly lines to make up for the inability of existing Government programs to supply equipment in the timeframe of the war. In one notable case, the Department of Commerce was asked by the manufacturer to help choose among Government agencies demanding its portable transceiver.

Second, if the assembly line to the Fleet deployment schedule is paced and if the Cyclic Acquisition engine to produce incremental improvements is fueled, both the operational tempo and the tempo of technological change in industry are taken into consideration. The result is that we can inject state-of-the-art technology systematically and cheaper with each deploying force, group, and squadron. This has been the approach taken in OP-094 for the last 30 months, and it lends itself to much broader use because the variable in the equation is the generational change of technology not the technology itself. So, whether weapons or microchips—or microchips in weapons—we can field equipment by "Builds."

Builds

"Builds" we define as fielding increments of a Group that are defined in terms of the length of a technological generation and the operational tempo of the platform. As either increases, the Build length shortens, and the Pyramidal Program structure is adjusted. As either decreases, a similar change is initiated. This construct is also useful in transitioning the Cold War programs of enduring value into the post-Cold War.

Figure 18 shows a series of builds that might be constructed for a program like JTIDS. JTIDS was a new idea in 1974; its current funding level will put its FOC in Navy in 2003. During those three decades, at eighteen months per generation, electronics will have gone through seven generations. If we do nothing, at the end of that time, when electronics will be cheaper to buy, cheaper to operate, and far more capable; JTIDS will cost more and do less than it should.

However, a Cyclical Acquisition engine can feed the assembly line with new building blocks with each Build. Starting at Build 1, we would field the existing system on deploying battle groups and begin prototyping Build 2 immediately recognizing that Build 1 is substandard. In the example in the figure, the prototype is ready and inserted at Build 3 as the JTIDS second generation (approximately two years from Build 1 on current carrier deployment schedules.) Although training will be somewhat different, so long as attention is paid in the design of the prototype to make its interfaces compatible with earlier builds, the first generation JTIDS carriers are at no disadvantage.

Build 3 JTIDS in the figure—continually monitored by the four focal points on the Cycle, may be technologically refreshed (e.g., a new board, software upgrades) during Build 4 or Build 5. Build 6, which might be on

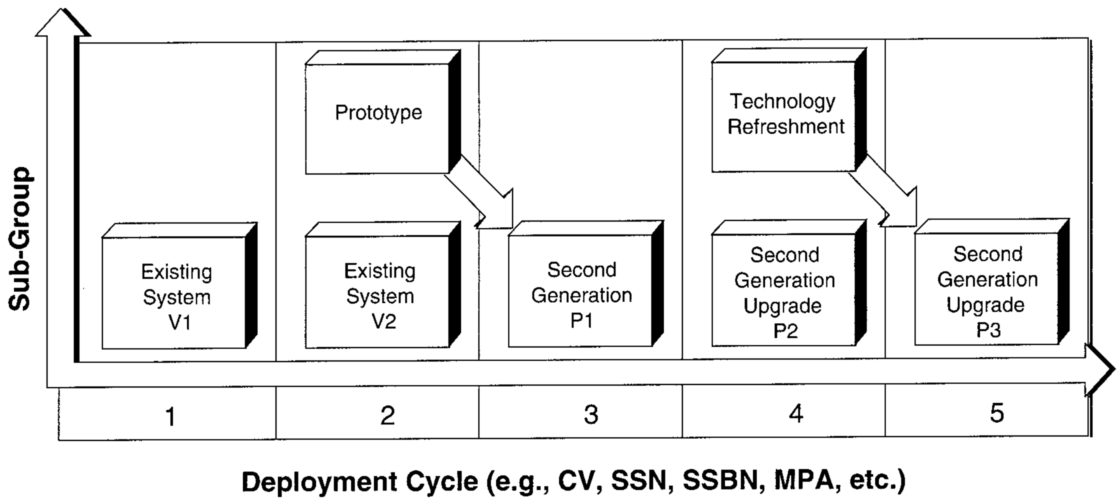


Figure 18. Assembly Line Builds

breadboard during Build 4, could represent a third generation JTIDS—and so.

Blocks

Third, and finally, Builds lend themselves to implementing new support strategies for logistics, for training, and for maintenance. Figure 19 shows the two types of support envisioned: component support and system support. Component support is aimed at the specific building block produced from the Cycle and reflects recognition that support for an antenna should be different than support for a computer. At the bottom of the Croesus Pyramid, where the number of building blocks are articulated, just as we foresee families of building blocks (e.g., TAC-3, KG-84, EHF terminals), we can also foresee families of "support blocks" that reflect technological families (e.g., computers, antennas, modems, radios.) Farther up the pyramid at the pillar level, however, system support is also required—for example, software maintenance or system troubleshooting for a GLOBIXS or within a TCC will be needed.

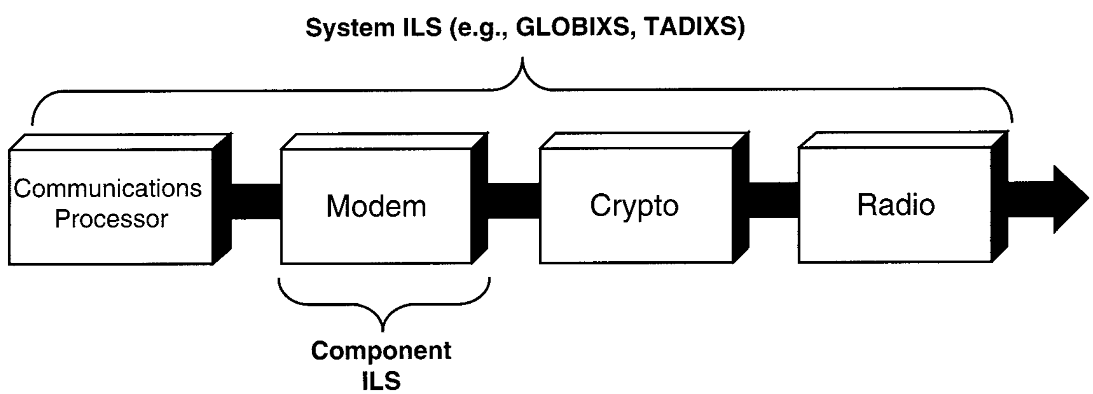


Figure 19. System Versus Component ILS

GLOSSARY

ACS	Afloat Correlation System
AR	Acquisition Requirement
ASWOC	Anti-Submarine Warfare Operations Center
BIPS	Billion Instructions Per Second
C4I	Command and Control, Communications and Computers, and Intelligence
CCC	CINC Command Center
CINC	Commander-In-Chief
CMST-N	Collection Management Support Tool-Navy
COMINT	Communications Intelligence
CWC	Composite Warfare Commander
DoD	Department of Defense
EHF	Extremely High Frequency
ELINT	Electronic Intelligence
ENWGS	Enhanced Naval Warfare Gaming System
EWCM	Electronic Warfare Coordination Module
FHLT	Force High Level Terminal
FIST	Fleet Imagery Support Terminal
FLTCINC	Fleet Commander-In-Chief
FOC	Fleet Operational Capability
FPC	Fleet Planning Center
GLOBIXS	Global Information Exchange Systems
HF	High Frequency
ILS	Integrated Logistics Support
IOC	Initial Operating Capability
IPS	Instruction Per Second
IRAD	Industrial Research & Development
JIT	Just in Time
JOTS	Joint Operational Tactical System
JTIDS	Joint Tactical Information Distribution System
MIPS	Million Instructions Per Second

MOE	Measures of Effectiveness
MPN	Manpower Personnel, Navy
NATO	North Atlantic Treaty Organization
NIPS	Naval Intelligence Processing System
NPDM	Navy Program Decision Memorandum
NTCS-A	Naval Tactical Command System Afloat
O&M,N	Operations and Maintenance, Navy
OBU	Ocean Surveillance Information System Baseline Upgrade
OPN	Other Procurement, Navy
OR	Operational Requirements
OSS	Operations Support System
POM	Program Objective Memorandum
POST	Prototype Ocean Surveillance Terminal
RDT&E	Research Development, Test and Evaluation
SCI	Special Compartmented Intelligence
SEW	Space and Electronic Warfare
SR	Support Requirement
STT	Shore Targeting Terminal
SYDP	Six Year Defense Plan
TACINTEL	Tactical Intelligence
TADIX	Tactical Data Exchange Systems
TCC	Tactical Command Center
TEMP	Test and Evaluation Master Plan
TFCC	Tactical Flag Command Center
THIPS	Thousand Instructions Per Second
TOA	Total Operational Availability
TOR	Tentative Operational Requirement
TQM	Total Quality Management
TR	Technical Requirement